1. Queen moves. This trivial program was hacked from QUEEN, which comes with the Stanford GraphBase system. It creates a graph of $n \times n$ vertices, representing the cells of a chessboard; two cells are considered adjacent if you can get from one to another by a queen move. Then it stores the graph as file `queen`*n*`x`*n*`.gb`. Other programs can therefore obtain a copy of the queen graph by calling *restore_graph*(`"queen`*n*`x`*n*`.gb"`).

```
#include "gb_graph.h"     /* we use the GB_GRAPH data structures */
#include "gb_basic.h"     /* we test the basic graph operations */
#include "gb_save.h"     /* and we save our results in ASCII format */
  long n;     /* the command-line parameter */
  char buf[100];
  main(int argc, char *argv[])
  { Graph *g, *gg, *ggg;
    ⟨ Process the command line 2 ⟩;
    g = board(n, n, 0L, 0L, −1L, 0L, 0L);     /* a graph with rook moves */
    gg = board(n, n, 0L, 0L, −2L, 0L, 0L);     /* a graph with bishop moves */
    ggg = gunion(g, gg, 0L, 0L);     /* a graph with queen moves */
    sprintf(buf, "queen%ldx%ld.gb", n, n);
    save_graph(ggg, buf);     /* generate an ASCII file for ggg */
    return 0;     /* normal exit */
  }
```

2. ⟨ Process the command line 2 ⟩ ≡

```
  if (argc ≠ 2 ∨ sscanf(argv[1], "%ld", &n) ≠ 1) {
    fprintf(stderr, "Usage:␣%s␣n\n", argv[0]);
    exit(−1);
  }
```

This code is used in section 1.

3. Index.

⟨ Process the command line 2 ⟩ Used in section 1.

QUEEN-GRAPH